

---

State of Minnesota, by Michael Campion,  
its Commissioner of Public Safety,

Court File No. 0:08-cv-00603-DWF-AJB

Plaintiff,

v.

**DECLARATION OF  
THOMAS E. WORKMAN, JR.**

CMI of Kentucky, Inc.,  
A Kentucky Corporation,

Defendant

Robert J. Bergstrom, Craig A. Zenobian,  
Shane M. Steffensen and Christopher D.  
Jacobsen,

Applicants

---

STATE OF MASSACHUSETTS	}	
	}	SS.
COUNTY OF BRISTOL	}	

I, Thomas E. Workman Jr., under oath, do depose and testify as follows:

1. I am an attorney, licensed to practice law in the Commonwealth of Massachusetts, with an office at 41 Harrison Street, Taunton, Massachusetts. I have been admitted to practice law in the Commonwealth of Massachusetts, before the United States Patent Office, and before the United States First Circuit.
2. I operate a computer forensics business, to which I dedicate most of my time and effort. This business offers expert analysis and testimony which deal with computers and the internet, and includes analysis and testimony

concerning computers that measure alcohol in the breath of human subjects, such as the Intoxilyzer 5000EN used in Minnesota.

3. The current version of my Curriculum Vitae is available on the internet at: <http://www.computers-forensic-expert.com/CV.PDF>, and has been printed and attached to this Affidavit.
4. I have published on the subject of source code and breath testing devices, and two of those publications are available to the public on the internet.
5. The first published paper, which I incorporate by reference to this affidavit, is available from the proceedings of the International Council on Alcohol Drugs and Traffic Safety, an organization with a website at: <http://www.icadts.org/>. My paper was presented at ICADTS 2007 in Seattle in August of 2007, and is published on the internet at: <http://www.icadts2007.org/print/196sourcecode.pdf>.
6. The second published paper, which I incorporate by reference to this affidavit, is available from the Suffolk Law School Journal of High Technology Journal, [http://www.law.suffolk.edu/highlights/stuorgs/jhtl/docs/pdf/Workman\\_Article.pdf](http://www.law.suffolk.edu/highlights/stuorgs/jhtl/docs/pdf/Workman_Article.pdf) and cited as 8 J. HIGH TECH. L. 209 (2008). The Journal has a website at: <http://www.jhtl.org/>.
7. I have reviewed the following materials:
  - a. October 28, 2008 Affidavit of Emerald Gratz;
  - b. October 28, 2008 Affidavit of Toby Hall;
  - c. October 28, 2008 Affidavit of Mario Santana;
  - d. October 28, 2008 Affidavit of William McNabb and attachments thereto;
  - e. June 6, 2008 Declaration of Charles A. Ramsay and attachments thereto;
  - f. The Complaint in this matter;
  - g. The answer to the complaint, and the counterclaims filed by CMI, all in this matter;
  - h. Plaintiff, Michael Campion, Commissioner of Minnesota Department of Public Safety's Memorandum in Support of Motion of Joint Decree and Order for Permanent Injunction;
  - i. Defendant CMI's Memorandum in Support of Motion of Joint Decree and Order for Permanent Injunction;
  - j. Proposed Consent Judgment and Permanent Injunction;

- k. Proposed Mutual Release and Settlement Agreement;
- l. Proposed Protective Order;
- m. Proposed Non-Disclosure Agreement;
- n. State of Minnesota Department of Administration Request for Proposal Titled: Evidentiary Breath Alcohol Test Instruments with opening date of October 28, 1996;
- o. Defendant CMI's October 25, 1996 unqualified acceptance of the State of Minnesota's Request for Proposal;
- p. September 26, 2006 e-mail between the Minnesota Department of Public Safety, Bureau of Criminal Apprehension and Defendant CMI;
- q. Testimony of Toby Hall in Tucson Arizona before Judge Bernini (I was personally present for Mr. Hall's direct testimony in Court and have reviewed the complete transcript of the direct examination and cross-examination);
- r. Intoxilyzer 5000 test result for Catherine P. Winters;
- s. Intoxilyzer 5000 Operator's manual;
- t. Affidavit of Karin Kierzak
- u. Affidavit of David Edin

### **What is Source Code**

- 8. The purpose of this affidavit is to provide the United States Federal District Court an overview of my anticipated testimony pertaining to the breath test software which controls the Intoxilyzer 5000EN, and more specifically the resolution of the dispute between the Attorney General of Minnesota and CMI, regarding the source code for the Minnesota Intoxilyzer 5000.
- 9. The Intoxilyzer 5000 is a device that contains an embedded computer system. An embedded computer system is a system that has a single specialized function, and does not incorporate an operating system, such as Windows or DOS.
- 10. An embedded computer system consists of hardware, firmware, and software.
- 11. The hardware component of the Intoxilyzer 5000 consists of all of the electrical and mechanical parts contained within the metallic chassis of the device, and includes the computer and memory associated with the computer.

12. The Intoxilyzer 5000 utilizes a Z80 microprocessor, originally manufactured by Zilog corporation, and introduced in the 1980s.
13. The firmware in the Intoxilyzer 5000 is represented by EPROM semiconductor devices, memory that can be erased and reprogrammed to contain new revisions of the software.
14. The software in the Intoxilyzer 5000 exists in three forms: source code, object code, and machine code.
15. Machine Code is the information that is loaded into the memory chips that reside inside every Intoxilyzer 5000, and are utilized by the computer semiconductor device within the Intoxilyzer 5000 in order to instruct the computer as to the discrete actions that the computer should take in executing its program, and thereby conducting a breath test or some other administrative task.
16. Object Code is sometimes referred to in the industry as “relocatable binaries” because they represent the information that relates to a single module or a single group of programmed functions. They differ from Machine Code in that they are pieces of the machine code, which when put together and modified to take into account precisely where in the continuum of instructions they reside, may become part of the Machine Code. The Object Code is processed by a program called a “Linker” and generates Machine Code.
17. When put together with the same set of software tools and a set of instructions commonly referred to as a “Makefile”, a set of Object Code will always be put together the same way and will always produce the same Machine Code.
18. A Linker will assemble only those pieces of object code that are required to satisfy all of the functions used by a program, and it is commonplace in the industry for the Machine Code to contain less than all of the instructions contained in the Source Code. This often occurs when some library routines are not needed, or when a part of the program has been rewritten, and some parts are no longer needed.
19. Source Code is the collection of programs and files which, when the various files are processed by programs referred to as Assemblers and Compilers, will produce Object Code that is in a form that can be combined in order to create the Machine Code which is loaded into every Intoxilyzer 5000.
20. When using the same set of software tools, and a sound platform for those tools, the same Source Code will create the same Object Code, and that Object Code when combined with the same Makefile will create an exact duplicate of the Machine File in the machine.

## **The Minnesota Source Code**

21. The Minnesota Model of the Intoxilyzer 5000EN, like every other recent instrument designed to test for alcohol in the breath of a human subject, is controlled by a computer, operating under the control of software.
22. Software is represented by the source code, the set of instructions and procedures, designed to be interpreted by the computer inside the breath test machine, that implement the science as set forth in the specifications.
23. The Minnesota source code is translated into machine code using specialized computer programs called compilers, assemblers and loaders, to produce machine code that is typically delivered in a specialized memory device referred to as an EPROM.
24. All non-trivial software has defects. When the computer executes those portions of the software that are defective, the machine that is controlled by the software often malfunctions. Those malfunctions, or “faults”, sometimes produce failures.
25. In breath testing machines, those failures can cause unexpected results: high BAC readings, unexplained readings, sample volume irregularities, and false reports that the defendant refused, because the Intoxilyzer erroneously believes that the citizen failed to provide an adequate sample of their breath.
26. Unlike hardware components that ultimately wear out and fail, a software defect is always and forever present, in every machine that contains the faulty software. Software failures are distinguished from hardware failures in that every machine that contains the faulty software is defective per se.
27. Since a basic premise is that the machines are properly designed, a presumption exists that if two machines produce the same results, the results must be correct, when in fact a software defect may produce identical, but wrong, results in every machine that is used.
28. In the United States, criminal prosecutions under DUI law are dominated by the various states, so it is not surprising that the challenges and litigation concerning the source code for evidentiary breath test machines have varied from state to state.
29. The greatest litigation activity seeking production of the source codes in various forensic breath instruments has occurred in Florida, New Jersey, and Arizona. I have participated in litigation concerning the operation of the Intoxilyzer in all of these states.
30. In *State v. Underdahl*, Minnesota recently unveiled a very favorable ruling (for the criminal defense bar). Several Georgia cases regarding source code

for the Intoxilyzer 5000 are making their way up to the appellate courts in that state.

31. Brief reports concerning the status of litigation in the three major states are included in the first paper that I published, which is referenced in this affidavit and attached to this affidavit. The reports for Florida, New Jersey and Arizona were compiled by Tom Hudson of Florida, Evan Levow of New Jersey, and Cliff Granger of Arizona.
32. A proper understanding of how software operates and is constructed, and in particular its application in Minnesota's Intoxilyzer 5000EN breath test machine, is important so that the Court can appreciate how any given machine operates and the role the source code plays in the operation of the Intoxilyzer.
33. When Source Code is converted to Machine Code, any ambiguities are either resolved (sometimes with "Warnings") or the Compiler that performs this translation documents "Fatal Errors" and refuses to produce the Machine Code.
34. Unlike natural languages, like English, computer languages operate on rigid syntax rules, and unlike spoken languages, they do not permit an interpretation based on the context of a communication. With software, each program step is rigidly interpreted, according to a strict set of syntactical rules, which sometimes create unintended results
35. The programmer who writes the source code is usually not the scientist upon whose science the machine is based, and is rarely the person who designed the hardware. Rarely does one person have all of the skill sets required to design, build, and program a machine.
36. The programmer works from a written specification, and often a "Systems Analyst" is employed to work with the author of the specification (the scientists and hardware designers) to write a detailed specification that is similar in concept to a blueprint for a building. With each handoff, opportunity for misunderstanding and mistakes in the final product, the source code, increases thereby degrading the quality of the product.
37. As programs become larger, they tend to be managed by multiple programmers, and just as a legal pleading that is constructed by multiple attorneys has a different degree of difficulty to manage, so too does larger software have its own peculiar problems. The more people involved, the greater the opportunity for miscommunication, increased complexity, and mistakes.
38. Each unique combination of software that is released to a customer in a manufactured or upgraded machine is a "software release". There exist

accepted procedures to define what is incorporated in a release, how it is tested, and how it is managed with respect to installation in the field.

39. Programmers often maintain and modify their source code months or years later, and if they are experienced, they add documentation in their source code, referred to as “comments”. These comments are ignored by the compilers that convert the source code to machine code. They provide context and logic as to why and how the source code is written.
40. Comments often express the programmer’s suspicions about problems that have been elusive and have not been specified and may not have been corrected. They may also contradict the instructions to the computer, as represented in the source code, meaning that either the comments are wrong, or the software is not correctly written.
41. An examination of the source code will often reveal extra steps that are not necessary to the computation of the results, but which will record information that relates to actual or potential error conditions. The mere existence of this kind of activity suggests that the programmers are trying to collect additional information in order to resolve problems they have seen, but have been unable to isolate and fix. In other instances, programmers have observed suspicious performance of the equipment during quality testing, and collect information to assist them in demonstrating the cause of the problems they observe.
42. The computer processor in the Breath Testing Machine will not operate correctly, and the software will produce unexpected results, if the environment is not controlled within the specifications.
43. The “environment” includes temperature, humidity, contaminants in the air, radio frequency interference (RFI), electromagnetic interference (EMI), and “dirty” electrical power supply sources.
44. Some deviations in the environmental specifications may damage the hardware so that it can no longer properly execute the instructions set forth in the source code, thereby creating permanent malfunctions which are not corrected when the environment is restored. Lightning strikes are one example of a phenomenon that delivers an unexpected electrical power surge that damages the internal electrical circuits, and probably causes a malfunction of the software that is operating at the time of the power surge caused by the lightning strike.
45. Defects may present themselves when an interferant is present, when the breath test machine’s pressure sensor detects parameters that are out of limits, or when some calculated result is deemed to be outside of accepted limits. When the state of the machine is such that a mistake in the source

code is encountered, and acted on, it is referred to as a “bug” in the software, and an unexpected result, or a “fault” occurs.

46. Faults may create results that are incorrect, perhaps logged or communicated to the operator, or they may be “handled” by exception routines in the software that are designed to “deal with” the faults, by taking some predetermined action to block the fault.<sup>1</sup> Fault-tolerant equipment is often designed to produce correct results in spite of faults, by making alternative calculations or by ignoring unreasonable data. If the computer is aware that a fault has occurred, it is common to log that fault so that a technician can later diagnose and fix the problem.

### **Reverse Engineering Source Code**

47. There exists a class of software tools called “Dis-Assemblers” which will accept as input a file that contains the Machine Code, and will produce as output a single file that contains assembly language instructions which could be processed in order to reinvent the Machine Code.
48. Dis-assemblers are used when the source code is not available. For example, when a virus is investigated, the machine code that is infected is often “Dis-assembled” so that a human can examine the instructions and understand the mechanism by which the virus operates.
49. Dis-assemblers are also used as part of an obfuscation strategy, in order to make it difficult to understand the program and its instructions.
50. There also exists a class of software tools which will obfuscate information and make it difficult to read. The first paragraph of this section has been reproduced here without any spaces or punctuation, and with the characters very close together, so as to make it extremely difficult to read the information. Yet all of the letters are there, in exactly the correct order. This is an example of what an obfuscation tool does to make it difficult to understand paragraph 47 of this affidavit.

Thereexistsaclassofsoftwaretoolscalleddisassemblerswhichwillacceptasinputafilethatcontainsthemachinecodeandwillproduceasoutputasingelfilethatcontainsassemblylanguageinstructionswhichcouldbeprocessedinordertoreinventthemachinecode

---

<sup>1</sup> Most computer systems will interrupt the flow of the software when known error conditions are detected, e.g. dividing by zero or attempting to access a memory location that does not exist. These error situations can be dealt with by ignoring the error, or by taking some predetermined action. By definition, there is no way to present a “correct” result. These errors indicate that the results from this execution of the software are invalid, yet many systems will deliver some answer “in spite of” knowing that the calculations are defective per se.

51. There are dozens of software tools that have been designed to accept as input the source code for a program, and will examine that source code and point out areas which may be mistakes or defects. These tools work well with genuine source code, and are ineffective when recreated source code, generated from a Dis-Assembler, is utilized.
52. The production of source code derived from a dis-assembler, and printed on paper, can only be characterized as a strategy to prevent a meaningful analysis of the source code, by disabling the ability of the reviewing entity to utilize the software tools that are commonplace in the industry, and which are designed to facilitate source code review.
53. When the source code for the New Jersey breath test machine was independently reviewed by two teams of experts, working independently of one another, both teams utilized the electronic and native formats of the source code to conduct their review.
54. I supervised the review of the source code in the State v. Jane Chun case in New Jersey, and I am familiar with the process by which the source code was reviewed.

### **Reviewing Source Code**

55. The Court should be informed that it is practically impossible to write a source code for any breath-test machine, including the Intoxilyzer 5000EN, which would test for all possible failures and then automatically correct them. The practical effect of this information is that errors are unavoidable and failures for these Intoxilyzers are occurring now.
56. A program that tests for all possible failures and then automatically corrects them *is not possible* because, as can be seen below, the mathematics of the “possibilities” are almost endless.
57. As to testing everything, to accomplish this, one must exercise every path in the software, where a path is a sequence of instructions that are used to instruct the processor.
58. By way of a simple example, suppose that a 1,000 line program contains an instruction which can alter the flow of what is next (a control statement, in the computer science language) as every 20th instruction. This would result in 50 control instructions,<sup>2</sup> which in their simplest form would permit two options for each control statement

---

<sup>2</sup> If every 20th instruction is a control statement, and there are 1,000 statements total, then there are 1,000 divided by 20 control statements, or 50 control statements.

59. For every control statement added, we double the number of paths possible. We express the number of paths as two raised to the nth power. For a program with 50 independent control points, the number of combinations is  $2^{50}$ , which creates 1,125,899,906,842,620 unique paths.
60. If we assume that we require one minute to initiate each unique combination and to check the results of each unique path through the software, then we need over 2 billion years<sup>3</sup> for one person to complete the testing, or if we could enlist every person in the United States to work on this task, we could complete it in a little over seven years of continuous work, 24 hours a day, seven days a week.
61. Just as proofreaders have developed lists of common errors in written works, so too have software quality engineers compiled lists of common errors that programmers often make while writing source code, as well as specific programs designed to test source code.
62. A computer scientist, preferably with a background in verifying software quality, may find defects in software by manually “reading” the source code. However, it is only possible, given the constraints of life expectancies and availability of such experts, to automatically review source code.
63. Whether automated or performed by a team of people who read the source code listings, this process is referred to as a “code review”. When a DUI-DWI defense attorney requests the source code for a Breath Testing Machine, it is with an aim to conduct a “code review” to look for defects.
64. Realistically, my opinion is that it would take one expert approximately 100 days to review the source code to the Intoxilyzer 5000EN were they able to utilize “automated” review of the source code, and provided that the correct and complete source code was provided in its native electronic format.
65. However, were that same expert to examine the source code manually, as under the terms of the Consent Judgment, that time spent “reading” the source code would increase by a factor of one-hundred; time spent reviewing code would escalate from 1/3 to 1/2 of a year to nearly 30 years.

**Current Source Code Litigation Against CMI  
is Proceeding on an Ad Hoc, State-by-State basis**

66. Source code litigation is not a recent phenomenon. Criminal defense attorney Marjorie Tedrick of Auburn, Washington made discovery inquiries about the BAC Datamaster software in the late 1980s. The breath testing technology of that era and the accessible software and codes of the device

---

<sup>3</sup> The number is computed by dividing  $2^{50}$  by 60 to compute the number of hours; divide that by 24 to get days; divide that by 365 to get years – the result is 2,142,123,110 years.

were fairly primitive compared to the current generation of devices in use across America today. Dr. A.W. Jones, of Linköping, Sweden calls today's machines, the "seventh generation" of forensic breath testing devices.

67. Challenges to breath testing and calibration activities of the Arizona breath test program began in earnest in the late 1990s. Elimination (by State employees) of "adverse" machine readings on an Intoxilyzer 5000 were "covered up" in order to make the device seem to work flawlessly. The fact that important computer data was being erased with the punching of a few computer keys triggered the wave of discovery motions and production orders that lead to the State of Arizona becoming the "Paul Revere" of criminal defense attorneys specializing in challenges to computerized forensic breath analyzers.
68. Florida followed closely behind Arizona with many crusaders seeking to prove that "conviction by computer" was un-American. The favorable court rulings and admissions by the State of flawed computer software in the Intoxilyzer 8000 series continued until the Florida legislature passed a new law that basically legislated the scientific correctness of the Intoxilyzer 5000, declaring the machine to be "perfect" by the stroke of Governor Bush's pen.
69. In Miami, in mid-October of 2008, a state inspector of Intoxilyzers was fired for falsifying inspection records, specifically for removing power from machines so that failed inspections were not recorded and reported.
70. In Houston, this week, an inspector of Intoxilyzer 5000 machines was arrested for falsifying inspection documents that relate to the Intoxilyzer.
71. In Florida and in Texas, as well as in Minnesota, the simple safeguards in the design of the source code are missing so that those with less than the best intentions can and do falsify information relating to the operation of the Intoxilyzer machines.
72. Arizona currently uses the Intoxilyzer 8000 for evidential breath alcohol testing. The regulatory agency is the Arizona Department of Public Safety [DPS], having taken over by statute from the Arizona Department of Health Services [ADHS]. A close, on-going relationship exists between CMI and DPS. To begin, the only evidentiary breath alcohol testing devices approved for use in the state of Arizona are manufactured and distributed by CMI.
73. It is difficult to give an up-to-date summary of the status of source code litigation in Florida, for several reasons. The subject is complicated by contrary judicial rulings among Florida's different judicial circuits. The situation is also extremely fluid. Cases are being decided every month

which continuously change the legal landscape. Early cases in Florida yielded some success in obtaining orders for the production of the code.

74. In addition to the litigation going on in Florida courts, the courts of Kentucky have also been involved through actions filed by CMI's attorneys seeking to quash and court order for production from another state's tribunals. The Kentucky trial-level courts have been extremely protective of their home-state corporation, CMI, Inc., in refusing to enforce subpoenas duces tecum issued by Florida courts pursuant to the Uniform Act to Secure Witnesses. *See*, for example, **Robbins v. CMI, Inc.**, unpublished opinion of the Ky. Ct. of App. 2006 WL 3524467 (December 8, 2006) where a Florida attorney's attempt to procure the Intoxilyzer's "source code" was fought by the manufacturer. [Unpublished opinions shall never be cited or used as authority in any other case in any court of this state. See KY ST RCP Rule 76.28(4).] From the opinion: Accordingly, we hold that CMI's motion to set aside and declare void the January 28, 2005, order of the Daviess District Court was improperly filed in the Daviess Circuit Court. As a motion seeking to set aside a judgment must be filed in the court that rendered the judgment, the proper procedure was to file a CR 60.02 motion in the Daviess District Court. As such, we are of the opinion that the Daviess Circuit Court committed reversible error by setting aside and declaring void the January 28, 2005, order and subpoena/subpoena duces tecum of the Daviess District Court.FN7 However, we further hold that the Florida court has no jurisdiction to directly enforce the order or subpoena/subpoena duces tecum issued by the Daviess District Court directed against Gilbert, a Kentucky resident, under the Uniform Act to Secure Witness. Such enforcement, including sanctions, may only be made through the Daviess District Court.

75. In *House vs Commonwealth of Kentucky*, 2007-CA-000417, reported on the Kentucky state court website at <http://www.kycases.com/2008/02/disclosure-of-i.html>. The Kentucky Court of Appeals held:

"Because a flaw in the computer source code of the Intoxilyzer 5000 would be consequential to the accuracy of the reading intended to be relied upon by the Commonwealth, such evidence is relevant and admissible. Accordingly, requesting the computer code to test the verity [sic] of the readings produced by the instrument is not unreasonable. A subpoena may be quashed only upon a showing that compliance therewith would be unreasonable or oppressive. We do not believe the Commonwealth and CMI have made this showing. The burden of providing the information is minimal and the expense de minimis."

76. In March, 2007, the Sarasota court (in a number of combined cases) issued a subpoenas duces tecum to CMI, Inc., ordering it to produce the source code. At present, it is unclear how the courts will sanction CMI for ignoring the order. The Sarasota Court sidestepped the issues of interstate service of the subpoena by finding that, as a foreign corporation registered to conduct business in Florida, CMI was subject to direct subpoena through its statutory agent.
77. CMI has moved in Kentucky to quash the Florida subpoena, but it is questionable whether the courts of Kentucky have jurisdiction to quash a Florida subpoena. In any event, the production dates for the CMI subpoenas have come and gone, and motions for sanctions against CMI are pending.
78. The software bug was specifically mentioned by Judge Kimberly Bonner in her order in *State v. Fabian*, [Case No. 2006-CT-009733-NC] upholding the materiality of the source code and authorizing a subpoena duces tecum against CMI in March of 2007. On July 10, 2007, Judge Bonner issued a civil contempt order against CMI for non-production of the source code pursuant to the subpoena issued under her court order. The judge's order cites some of the facts:
- a. CMI did not produce the requested material.
  - b. CMI did not file an Objection or a Motion challenging this subpoena with this Court.
  - c. The Defendants filed a Verified Motion for Order to Show Cause asking for CMI to be found in contempt for failing to comply with this Court's Subpoena.
79. Judge Bonner held CMI in contempt and has assessed a DAILY penalty of \$3200.00. That penalty commenced August 1, 2007, and before adding statutory interest, the daily fine, as of today, exceeds \$1.5 million dollars. CMI appealed the contempt finding, in Florida, and the Florida appellate courts affirmed the contempt finding.
80. I have testified on over ten occasions in Georgia, and as a result there exist orders from Georgia judges which require that CMI provide the source code to Georgia attorneys so that the Intoxilyzer 5000 used in Georgia can be evaluated.
81. I have testified in Louisiana, and as a result a Trial court judge ordered CMI to produce the source code for the Intoxilyzer 5000 used in Louisiana.
82. I have testified in Massachusetts, and as a result the Commonwealth of Massachusetts was ordered to produce the source code for the Draeger 7110 MK-III C, a machine that competes with CMI for market share of breath testing devices.

83. In states where issues relating to passwords, the proposal to supply bound printed copies for review at CMI in Kentucky, the proposed trade secret status, and copyright defenses from production have been litigated, CMI and the state have lost on every contention. It is my belief that these issues are being presented in bad faith, in an effort to prevent a meaningful review, and that Judges upon hearing the facts and law from CMI's president (Toby Hall), and from experts for the defense, have not adopted any of the significant positions set forward in the proposed order, as presented jointly by CMI and the Minnesota Attorney General.
84. Most of the provisions in the proposed order do not deal with the resolution of the dispute as set forth in the complaint and counter-claims, but instead set forth restrictions on state litigants as they defend themselves in the Minnesota state courts.
85. It is understandable why the chief law enforcement officer would want to limit the rights of criminal defendants, and why the manufacturer of the machines used to convict those defendants would wish to avoid liability, which could include significant exposure if CMI intentionally withheld information that they knew resulted in false convictions of Minnesota citizens.
86. The liberty of Minnesota citizens is conspicuously missing with regard to the parties entering into an agreement regarding their liberty. Since attempts by Minnesota criminal defense attorneys to intervene were unsuccessful, in the event that hearings are held to listen to the viewpoints of these citizens, the Federal Court has an opportunity to listen to these views for the first time.

**The Attorney General's Office Must Be Aware that this Consent Judgment Hampers Source Code Review, and is even Aware of Actual Errors that Would Be Uncovered Through Effective Review of the Source Code**

87. CMI's Intoxilyzer devices have been discovered to render invalid sample results for motorists who have given breath samples on CMI's device as demonstrated by the case of Catherine P. Winters.
88. E-mail conversations I have reviewed demonstrate that the BCA has been aware for years that the Intoxilyzer 5000 can and will deem breath samples as "deficient" or not depending entirely upon which version of software the machine is running. This is the situation confronted by Catherine Winters.
89. The Intoxilyzer machines are particularly vulnerable to defects relating to miscalculation of breath volume, and of incorrectly rejecting breath samples that it knows are valid and correct. The source code in the Florida Intoxilyzer was directly responsible for such a situation in the past two

years, and only CMI has the detailed knowledge, which they will not share, as to why the defect existed in Florida, what they did to fix it, and what other states will experience the same defect.

90. In response, BCA scientists issued sworn affidavits, on behalf of the MN AG's Office, dismissing this evidence. These affidavits claim the manufacturer had satisfactorily addressed the problem.
91. In reality, additional e-mails were uncovered showing that this known software problem had *not* been fixed. CMI did try to correct the problem in April 2007 by providing the state with updated software. However, according to information provided to me, that update was never installed statewide. There can be no legitimate reason for failing to correct such a defect when a correction is known and implemented and ready to install.
92. This evidence leads me to conclude that at least one known software bug exists within the programming to the Intoxilyzer 5000, a bug that has gone deliberately uncorrected, resulting in wrong results for an indeterminate number of Minnesota citizens.
93. The documented, established invalid reported sample results from Catherine Winters (and others) from tests undergone on the Intoxilyzer 5000 device can be explained after a thorough forensic review of the source code to the Intoxilyzer 5000 is conducted.

**The Proposed Consent Judgment and Permanent Injunction Prohibits  
The Reasonable Access And Analysis Of The Source Code**

94. If the Court were to enter an order affirming the settlement agreement as proposed, and if state courts were to defer to the Federal Court as a court of superior jurisdiction, then the ability to identify the cause of problems that are known to exist in the Minnesota Intoxilyzer 5000's source code would be impaired to a point where defendants would be denied a reasonable opportunity to discover the extent of problems that exist in the source code.

**The Copyright basis for this lawsuit was never properly litigated, and the  
contract between the parties clearly transfers Copyrighted Or Copyrightable  
Materials Relating To The Intoxilyzer Device To The State of Minnesota**

95. The State of Minnesota Request for Proposal requires the transfer of ownership to the Plaintiff State of Minnesota any copyrighted or copyrightable material relating to the fleet of Intoxilyzer breath testing instruments that the State of Minnesota Sought through the Request for Proposal. Paragraph 29 of the Request for Proposal states:

All right, title, and interest in all copyrightable material which Contractor shall conceive or originate, either individually or jointly with others, and which arises out of the performance of this

Contract, will be the property of the State and are by this Contract assigned to the State along with ownership of any and all copyrights in the copyrightable material. Contractor also agrees, upon request of the State to execute all papers and perform all other acts necessary to assist the State to obtain and register copyrights on such materials. Where applicable, works of authorship created by Contractor for the State in performance of this Contract shall be considered “works for hire” as defined in the U.S. Copyright Act.

96. Paragraph 29 of the Request for Proposal is intended to transfer to the State of Minnesota ownership of all copyrighted or copyrightable material relating to the Intoxilyzer device the State of Minnesota obtained through the Request for Proposal.
97. The source code for the Intoxilyzer 5000 will not function on any competing company’s machines, and will not function correctly in any other state or country.
98. A provision transferring ownership of all copyrighted is common in the marketplace. Such provisions are designed to protect the purchaser, in this case, the State of Minnesota, from incidents where the manufacturer goes out of business, files for bankruptcy, or the ongoing product design and maintenance is involuntarily transferred to another party.
99. If the source code were merely licensed to the state of Minnesota, under a federal bankruptcy proceeding, a Bankruptcy trustee can and will set aside contractual agreements for the benefit of creditors, and could sell the copyright for the source code to the highest bidder at auction. Minnesota would then have to deal with a third party who could charge any amount for the continued development or correction of problems in the machine. Without such a transfer of ownership of the copyright, the State of Minnesota would be unable to use and adjust the intoxilyzer device to meet its needs should the manufacturer no longer exist.
100. Defendant CMI’s claim that portions of the source code predate the State of Minnesota’s Request for Proposal does not change the fact that CMI was required to transfer all copyrightable interest to the State of Minnesota. The right to create a derivative work is an enumerated right under the Federal Copyright statutes (17 USC section 106) that belongs to the copyright owner. The created derivative work, which is the Minnesota source code, enjoys the same copyright protection that a newly written source code work would enjoy.

### **The Source Code To The Intoxilyzer 5000 Is Not A Trade Secret**

101. CMI applied for and obtained a patent from the United States Patent Office for the Intoxilyzer 5000. A true and correct copy of the patent for the Intoxilyzer 5000 is attached to this declaration.
102. When applying for its patent for the Intoxilyzer 5000, Defendant CMI was required to fully disclose how the subject machine operated, so that a person having ordinary skill in the art could make and practice the invention upon the expiration of the patent 35 USC 112<sup>4</sup>.
103. While it was not necessary to disclose the source code to the patent office, it was CMI's duty to fully disclose how the machine operates, and that includes disclosing everything of significance that the Intoxilyzer 5000 did at the date of the application for patent.
104. CMI's public filing and its subsequently obtained patent establish the Source Code is or should be generally known or readily ascertainable and cannot be a trade secret.
105. The issue of Trade Secret status was litigated in a multi day hearing in Arizona, before Judge Bernini in Tucson, at which CMI President Toby Hall testified. The Court specifically held that the Intoxilyzer source code was not a trade secret.
106. Both Arizona and Florida Courts have held that CMI's source code in its Intoxilyzer devices is not a protected trade secret, because any person who observes the operation of a breath test can observe what the machine does, and in so doing can ascertain what the source code is instructing the machine to do. The single exception is the manner in which the machine calculates and reports the amount of alcohol in the citizen's breath.
107. The passwords for the Intoxilyzer are well known. If the passwords are contained in the source code, as CMI claims, such an irresponsible design would require the release of a new source code every time a state employee retired, was fired, or terminated employment for any reason – if that employee knew any of the passwords. Knowledge of the password must be accompanied by physical access to a machine, and only a person with both can utilize the information.

---

<sup>4</sup> 35 U.S.C. 112 Specification. The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same, and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Dated: October 31, 2008

/s/ Thomas E. Workman Jr.

Thomas E. Workman Jr.

41 Harrison Street

Taunton, MA 02780

508 822-7777